ECONSTOR

WWW.ECONSTOR.EU

Der Open-Access-Publikationsserver der ZBW – Leibniz-Informationszentrum Wirtschaft
The Open Access Publication Server of the ZBW – Leibniz Information Centre for Economics

Bottazzi, Giulio

Working Paper

Subbotools user's manual: For version 0.9.7.1, 8 September 2004

LEM Working Paper Series, No. 2004/14

Provided in Cooperation with:
Laboratory of Economics and Management (LEM), Sant'Anna School of Advanced Studies

Suggested Citation: Bottazzi, Giulio (2004) : Subbotools user's manual: For version 0.9.7.1, 8 September 2004, LEM Working Paper Series, No. 2004/14

This Version is available at:
http://hdl.handle.net/10419/89338

Nutzungsbedingungen:
Die ZBW räumt Ihnen als Nutzerin/Nutzer das unentgeltliche, räumlich unbeschränkte und zeitlich auf die Dauer des Schutzrechts beschränkte einfache Recht ein, das ausgewählte Werk im Rahmen der unter
→ http://www.econstor.eu/dspace/Nutzungsbedingungen
nachzulesenden vollständigen Nutzungsbedingungen zu vervielfältigen, mit denen die Nutzerin/der Nutzer sich durch die erste Nutzung einverstanden erklärt.

Terms of use:
The ZBW grants you, the user, the non-exclusive right to use the selected work free of charge, territorially unrestricted and within the time limit of the term of the property rights according to the terms specified at
→ http://www.econstor.eu/dspace/Nutzungsbedingungen
By the first use of the selected work the user agrees and declares to comply with these terms of use.

ZBW Leibniz-Informationszentrum Wirtschaft
Leibniz Information Centre for Economics

Mitglied der Leibniz-Gemeinschaft

SUBBOTOOLS User's Manual

For version 0.9.7.1, 8 September 2004

Giulio Bottazzi

Copyright © 2003-2004 Giulio Bottazzi

Permission is granted to make and distribute verbatim copies of this manual provided the copyright notice and this permission notice are preserved on all copies.

Permission is granted to copy and distribute modified versions of this manual under the conditions for verbatim copying, provided that the entire resulting derived work is distributed under the terms of a permission notice identical to this one.

Permission is granted to copy and distribute translations of this manual into another language, under the above conditions for modified versions, except that this permission notice may be stated in a translation approved by the Free Software Foundation.

Table of Contents

1 Introduction 1

2 Overview 2
- 2.1 The Subbotin Families of Densities 2
- 2.2 Parameters Estimation 3
- 2.3 Programs Input/Output 5

3 Tutorial 6
- 3.1 Command Line Tutorial 6
- 3.2 Graphic Tutorial 7

4 Generating Subbotin Random Variates 9
- 4.1 subbogen 9
- 4.2 subboagen 9

5 Fitting the Subbotin Density 10
- 5.1 Fitting the Subbotin density in the symmetric case 10
  - 5.1.1 Method of Moments 10
  - 5.1.2 Unconstrained Minimization 10
  - 5.1.3 Minimization on Intervals 10
- 5.2 Fitting the Subbotin density in the asymmetric case 11
- 5.3 Fitting the asymmetric Laplace density 11
- 5.4 subbofit 11
- 5.5 subboafit 12
- 5.6 subbolafit 13
- 5.7 laplaafit 14

6 Exploring the symmetric Subbotin Log-Likelihood 15
- 6.1 subboshow 15

7 Installation 16

8 License and Copyright 17

Concept Index 18

1 Introduction

The SUBBOTOOLS package is intended as an help to the use of the Subbotin family of probability densities in a statistical analysis environment.

The package contains various programs for the maximum likelihood estimation of an unknown distribution and for the generation of pseudo random variables. The program `subbofit` can be used for the "fitting" (estimation) of a Subbotin probability density on an user supplied dataset. The programs `subboafit` and `subbolafit` perform this estimation on two larger classes of probability densities: the asymmetric Subbotin densities (a 5 parameter family) and the so called "less-asymmetric" Subbotin densities (a 4 parameter family) (see Chapter 2 [Overview], page 2). The program `laplaafit` restrict the estimation to the family of asymmetric Laplace densities, a subset of the asymmetric Subbotin.

In addition, the programs `subbogen` and `subboagen` can be used to generate pseudo random variates drawn, respectively, form a user-specified Subbotin or asymmetric Subbotin density (see Chapter 4 [Subbotin Random Variates], page 9) while `subboshow` allows the exploration of the (symmetric) Subbotin log-likelihood function (for a definition see Section 2.2 [Parameters Estimation], page 3) of an user supplied dataset (see Chapter 6 [Exploring Subbotin Log-Likelihood], page 15).

All the programs in this package make use of functions from the GNU Scientific Library, a collection of numerical routines for scientific computing. These libraries must be installed on your system for SUBBOTOOLS to compile and run. More information about GSL can be found at the project home page, `http://www.gnu.org/software/gsl/`. Follow the instruction provided there to download and install the GSL library.

Using appropriate command line options, the programs `subbofit`, `subboafit`, `subbolafit`, `laplaafit` and `subboshow` can be used to produce "pictures". They can produce tabulated data, in ASCII format, representing 1-dimensional curves or 2-dimensional surfaces. The output structure, made of newline separated records of two or tree space separated numeric fields, is in a format suitable to be sent to the common plotting utilities ordinarily found on Unix systems. For instance, I usually use this "graphic" output options in conjunction with `Gnuplot` or the `graph` utility found in the `plotutils` package. For more information on the output format of these programs see Section 2.3 [Programs Input/Output], page 5.

SUBBOTOOLS has been developed on Linux systems. It is freely distributed under the GNU General Public License and is provided “as is” without any explicit or implicit warranty (see Chapter 8 [License and Copyright], page 17). In principle it should work in any flavor of Unix with minor or without modifications. No porting to different OS’s has been undertaken nor planned.

The last version of SUBBOTOOLS can be found at `http://bottazzi.sssup.it/~giulio`. Comments and bug reports are welcome. E-write to `giulio.bottazzi@sssup.it`.

Please read the ‘`INSTALL`’ file provided with the package for installation instructions and the file ‘`README`’ for an updated description of the package components.

2 Overview

The `SUBBOTOOLS` package contains the following programs:

`subbogen, subboagen`
: these programs generate random variable extracted from a symmetric (`subbogen`) or asymmetric (`subboagen`) Subbotin density. The parameters that defines the density can be provided in the program command line or read from standard input see Chapter 4 [Subbotin Random Variates], page 9.

`subbofit, subboafit, subbolafit, laplaafit`
: these programs find the symmetric (`subbofit`) and asymmetric (`subboafit`) Subbotin density or the asymmetric Laplace density (`laplaafit`) that better fit a given set of observations. The observations are considered independently drawn from the same, unknown, probability density. The values of the density parameters are estimated, depending of the family, with maximum likelihood or with the method of moments (see Chapter 5 [Fitting Subbotin Density], page 10).

`subboshow`
: this program takes a set of observations as input and produces a graphic showing the symmetric Subbotin log-likelihood of this set as a function of the density parameters see Chapter 6 [Exploring Subbotin Log-Likelihood], page 15.

All these programs deal, in a way or another, with one of the Subbotin families of densities. The symmetric Subbotin densities depend on three parameters, usually denoted m, a and b. The first parameter, m, represents the density "positioning" parameter, i.e. the position of its "center". Indeed it is at the same time the mean, the mode and the median of the density. The second parameter, a, is a "scale" parameter, that express the "spread" of the distribution. It is proportional (but in general not equal) to the density standard deviation. The last parameter, b, is a "shape" parameter. It tells how large are the "tails" of the density with respect to its center.

The asymmetric Subbotin densities extend the previous family by introducing two different values for the parameters a and b at the left and at the right of the density mode m. This extended group of densities constitutes a 5 parameters family. The parameters are commonly denoted b_l, b_r, a_l, a_r and m . In this case, the positioning parameter m represent the mode of the density, but it is no longer its mean nor its median.

The asymmetric Laplace family is a subset of the asymmetric Subbotin family obtained putting $b_l = b_r = 1$.

For a precise definition of the Subbotin, asymmetric Subbotin and asymmetric laplace families Section 2.1 [The Subbotin Families of Densities], page 2. For the estimation of the density that better fits a provided sample different methods can be applied. For a short discussion of the problem Section 2.2 [Parameters Estimation], page 3. The strategy of the estimation procedures implemented in the programs distributed with this package is described in Chapter 5 [Fitting Subbotin Density], page 10.

The program of this package share a common procedure to read in input and print out output. Input is read from standard input in ASCII format and output is sent to standard output, always in ASCII format. For more details, see Section 2.3 [Programs Input/Output], page 5.

2.1 The Subbotin Families of Densities

The functional form of the Subbotin density is characterized by three parameters, a positioning parameter m, a scale parameter a and a shape parameter b, and reads

$$f(x;a,b,m) = \frac{1}{2\,a\,b^{1/b}\,\Gamma(1+1/b)}\; e^{-\frac{1}{b}\left|\frac{x-m}{a}\right|^{b}} \quad .$$

Due to its symmetry, the Subbotin density has all central moments of odd order equal to 0. The central moment of order $2l$ reads

$$M_{2\,l} = \left(a\, b^{1/b}\right)^{2\,l} \; \frac{\Gamma((2\,l+1)/b)}{\Gamma(1/b)}$$

while the absolute deviation is

$$M_{|1|} = a\, b^{1/b} \; \frac{\Gamma(2/b)}{\Gamma(1/b)}$$

The asymmetric Subbotin density extends the family described above by considering different values for the parameters a and b in the two halves of the density. Its functional form depends on five parameters: a positioning parameter m, two scale parameters a_l and a_r respectively for the values below or above m, and two shape parameters b_l and b_r characterizing, respectively, the lower and upper tail of the density. The functional form reads

$$f(x;a,b,m) = \begin{cases} \frac{1}{A} e^{-\frac{1}{b_l}\left|\frac{x-m}{a_l}\right|^{b_l}} & x < m \\ \frac{1}{A} e^{-\frac{1}{b_r}\left|\frac{x-m}{a_r}\right|^{b_r}} & x > m \end{cases}$$

where

$$A = a_l b_l^{1/b_l}\Gamma(1+1/b_l) + a_r b_r^{1/b_r}\Gamma(1+1/b_r)$$

It is rather straightforward to obtain the expression of the central moments of the asymmetric Subbotin density. They are in general all different from zero, the density being skewed. Since we are not going to make use of the central moments for the asymmetric case, their expressions are not reported here.

The asymmetric Laplace density restricts the family described above by considering fixed values for the tail parameters $b_l = b_r = 1$. Its functional form depends on three parameters: a positioning parameter m and two scale parameters, a_l and a_r, for the parts of the density below or above m, respectively. The functional form reads

$$f(x;a,b,m) = \begin{cases} \frac{1}{A} e^{-\left|\frac{x-m}{a_l}\right|} & x < m \\ \frac{1}{A} e^{-\left|\frac{x-m}{a_r}\right|} & x > m \end{cases}$$

where

$$A = a_l + a_r$$

2.2 Parameters Estimation

Consider a set of N observations $\{x_1, \ldots, x_N\}$ and assume that they are independently drawn from an unknown Subbotin distribution. We are interested in the estimation of the parameters (a, b, m) of this unknown distribution .

A first approach to this problem, commonly known as *method of moments*, is based on the comparison between the theoretical moments of the density (see Section 2.1 [The Subbotin Families of Densities], page 2) and the sample moments computed starting from the N observations.

The procedure goes as follows. We compute the sample mean M_0 , the sample variance M_2 and the sample absolute deviation $M_{|1|}$. First, we obtain an estimation of the central parameter of the distribution using the sample mean. We set $m = M_0$.

Second, we obtain the estimation for the parameter b as the unique root of the equation

$$\frac{\Gamma(3/b)\Gamma(1/b)}{\Gamma(2/b)} = \frac{{M_{|1|}}^2}{M_2}$$

and finally the estimated value for the parameter a is given by

$$a = b^{-1/b}\sqrt{\frac{M_2\Gamma(1/b)}{\Gamma(3/b)}} \quad .$$

A second possible approach consists in the maximization of the empirical likelihood, i.e. in the method generally known as *maximum likelihood estimation*. More specifically, instead of maximizing the likelihood of the sample, we minimize the negative log-likelihood, computed taking the logarithm of the likelihood function and chancing it sign. Moreover, using the first order conditions is easy to derive the expression of the estimated value of the parameter a as an explicit function of b and m. Substituting this expression in the negative log-likelihood one obtains an object function that depends only on 2 parameters, b and m. We call this function *reduced negative log-likelihood* and it reads

$$L(a,b) = \log(2b^{1/b}\Gamma(1+1/b)) + \frac{1}{b} + \frac{1}{b}\log(\frac{1}{N}\sum_{j=1}^{N}|x_j - m|^b) \quad .$$

In the case of the asymmetric density, the method of moments is not so attractive. Indeed it is impossible to obtain simple expressions relating the parameters characterizing the density to the various moments of the sample. Then the use of numerical methods becomes mandatory, and one is naturally led toward maximum likelihood estimation. The (negative) log-likelihood (per observation) in the case of asymmetric density reads

$$L(b_l, b_r, \ldots) = \log(A) + \frac{1}{Nb_l}\sum_{x_j<m}\left(\frac{m-x_j}{a_l}\right)^{b_l} + \frac{1}{Nb_r}\sum_{x_j>m}\left(\frac{x_j-m}{a_r}\right)^{b_r} \quad ,$$

$$A = a_l b_l^{1/b_l}\Gamma(1+1/b_l) + a_r b_r^{1/b_r}\Gamma(1+1/b_r) \quad .$$

As can be seen, both $L(a,b)$ and $L(b_l, b_r, ...)$ are not analytic functions of their arguments. This is way their minimization is a non-trivial task that must be handled with particular care, see Chapter 5 [Fitting Subbotin Density], page 10.

The situation is much easier for the case of the asymmetric Laplace density. Indeed by substituting $b_l=b_r=1$ in the previous expression one obtains a reduced log likelihood as a sole function of the mean m

$$L(m) = -2\log\left(\sqrt{S_l} + \sqrt{S_r}\right) - 1$$

where

$$S_l = \frac{1}{N}\sum_{x_j<m} m - x_j \qquad S_r = \frac{1}{N}\sum_{x_j>m} x_j - m$$

It is also possible to see that this one variable reduced log-likelihood does posses a single global minimum. The optimal parameters a_l and a_r are determined by the relations

$$a_l = S_l + \sqrt{S_l\ S_r} \qquad a_r = S_r + \sqrt{S_l\ S_r}$$

2.3 Programs Input/Output

Both `subbofit` and `subboshow` require user supplied data. The data can be read in ASCII format from files whose name are specified at the end of the command line (after any option) or from the standard input. Different data are separated by white characters (spaces or tabs) or newlines. Lines beginning with a fence symbol '#' are considered comments and ignored.

The output of `subboshow` and the output of `subbofit`,`subboafit`,`subbolafit` and `laplaafit` whith the options '`-O 1`' or '`-O 2`', are intended to be used to produce pictures. They consist in tabulated data, of 2 (for 2-dimensional plots) or 3 (for 3-dimensional plots) columns, where each line correspond to a different point. In the case of 3-dimensional plots, the triplet associated to different values of the first variable are separated by 2 newlines. This format has been chosen to provide an easy interface to the `gnuplot` plotting program. The 2-dimensional plots can be easily displayed also making use of the `graph` utility of the `plotutils` package. For an example of the use of the "graphic" output Section 3.2 [Graphic Tutorial], page 7.

The `gnuplot` program FAQ can be found at `http://www.gnuplot.info/faq/`.

For information about the `plotutils` package check `http://www.gnu.org`

3 Tutorial

This tutorial will drive you across some simple example on the use of the programs provided with this package. The tutorial is divided in two parts. In the first part the use of the programs from the command line will be illustrated. We will learn how to generate pseudo-random variables drawn from a Subbotin density and how to obtain the best Subbotin fit on a given dataset. The second part of the tutorial is more "graphical", and we will see how to use the programs in distributed with `SUBBOTOOLS` to generate plots inside the `Gnuplot` plotting environment. Even if I'm not going to assume a previous knowledge of `Gnuplot`, I will not explain all the details of the different commands that I will use. For more information on the different commands and switches, you are referred to the exhaustive help system of `Gnuplot` itself (try `help` after `Gnuplot` invocation).

Please, understand that the choice of `gnuplot` as plotting utility is due to my long experience with it, and to its proven reliability and easiness of use. The output of the `SUBBOTOOLS` programs has been designed according to the `gnuplot` requirement as illustrated in Section 2.3 [Programs Input/Output], page 5. Due to the essential looseness of these requirements, however, it should be relatively easy to use different utilities.

In this tutorial I'm going to suppose that you are familiar with the Unix shell (command line) interface and that you already installed the `SUBBOTOOLS` package without problems.

3.1 Command Line Tutorial

In order to avoid danger to possibly relevant files, before trying the following steps create a new directory, something like '`subbotest`', and move there

```
mkdir subbotest
cd subbotest
```

Now that you are in a more or less safe environment, let's start. We begin by generating a meaningful set of numbers to work with

```
subbogen -N 100 -m0 -a1. -b1. -R 5 > testdata.txt
```

Let's see the meaning of the previous command. Using `subbogen` we generated 100 random numbers from a Subbotin distribution with *m=0*, *a=1* and *b=1*, i.e. a Laplace (symmetric exponential) distribution centered in zero and with variance equal to 2. The numbers are generated initializing the RNG (random number generator) with a seed equal to 5 and are placed in a file named '`testdata.txt`'. One can generate independent samples by using the option '`-R`' with different integer numbers (the default value for the seed is 0).

The file '`testdata1.txt`' now just contains a column of 100 number, you can easily inspect it using an editor. We now use this file to investigate the properties of the program `subbofit`.

We begin with the simplest command

```
subbfit < testdata1.txt
```

that should generate an output similar to

```
8.832028e-01 8.777607e-01 2.917261e-02 1.615783e+00
```

The meaning of these four number is as follows: the first three number represent the estimated values for b, a and m respectively, while the last number is the (negative) log-likelihood associated with these three values. As can be seen, the estimated values for b and a are quite far from the real values. The situation is different for different seeds but with only 100 observations an error of about 10% is not uncommon.

In addition to the maximum-likelihood estimation of the parameters, which is the default, the program `subbofit` also implements a method of moments. You can choose the method with the command line option '`-M`', try

```
subbofit -M 0 < testdata1.txt
```

and you should obtain something like

```
9.569862e-01 9.104554e-01 5.291858e-02 1.617363e+00
```

In this particular case, the method of moments provides better estimates than the ones obtained via maximum-likelihood. Of course the associated log-likelihood is smaller. In order to obtain more details on the procedure of fitting you can use the option '`-V`'. For more details Chapter 5 [Fitting Subbotin Density], page 10.

Another interesting feature is the possibility of fitting just the b and a parameters, providing the "true" value of m directly from the command line. This is useful when you work with *normalized* data whose mean has been previously subtracted. Since this is a quite common situation `subbofit` provides the '`-m`' command line option. With the file generated above do

```
subbofit -m 0 < testdata1.txt
```

and you should obtain

```
8.808684e-01 8.777596e-01 0.000000e+00 1.616923e+00
```

and in this case the variation in the fitted parameters is relatively low.

3.2 Graphic Tutorial

Now let's start `gnuplot` and see how the programs in `SUBBOTOOLS` can be used inside its graphic environment. Simply type

```
gnuplot
```

and you will be greeted by a fairly large amount of informations and, finally, a prompt. As a first exercise let's compare the result of our fitting procedure with the original function. Using

```
gnuplot>set log y
gnuplot>plot "<subbofit -O 2 < testdata.txt" title "fit",\
>exp(-abs(x))/2 title "th."
```

you can produce a graph similar to the one in Fig.1

The option '`-O 2`' print for each point of the file '`testdata1.txt`' the value of the probability density associated with the fitted parameters. `gnuplot` interprets this output and produces the corresponding graph. With our small sample the obtained estimate is notably different from the theoretical Laplace density from which the data were originally generated.

A similar comparison can be performed on the distribution function using the command

```
gnuplot>set key top left
gnuplot>unset log y
gnuplot>plot "<subbofit -O 1 < testdata1.txt'' title "fit",\
>(x<0?exp(-abs(x))/2:1.-exp(-abs(x))/2) title "th."
```

where the option '`-O 1`' now prints the value of the estimated distribution function for each input points. The result you should obtain is similar to the one reported in Fig.2.

The program `subboshow` is conceived for the visual exploration the Subbotin "reduced" lok-likelihood. For a precise definition Chapter 5 [Fitting Subbotin Density], page 10. The program takes a set of observations and print out the associated value of the Subbotin reduced negative log-likelihood for a grid of b and m values. Using the previously generated observations, try the following

```
gnuplot>splot "< subboshow -b0.5:1.5 -m-.1:.11 <testdata.txt ",\
> i 0 notitle w l
```

and you will obtain a 3D plot representing the negative log-likelihood of the dataset similar to Fig. 3 .

If one is interested in the behaviour of the reduced (negative) log-likelihood with respect to a specific parameter it is enough to specify a single point for the other parameter. For instance, if one is interested in the behavior of the function with respect to the parameter b keeping fixed the value of m, one can use

```
gnuplot>plot "< subboshow -b0.5:1.5 -m0 -M 1 <testdata.txt ",\
> i 0 notitle w l
```

and obtain a 2D plot similar to Fig.4 .

The options '`-b`' and '`-m`' set the region for which points are generated. See the help of `subboshow` for more options.

Using this command, you can check if the parameters values previously found with `subbofit` do in fact constitute a minimum.

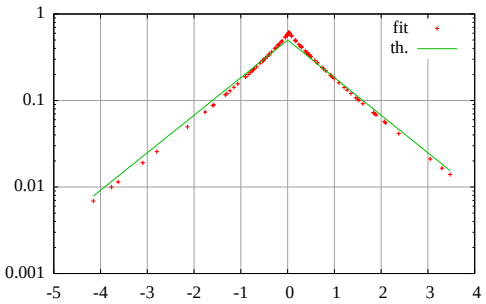

Figure 1: Estimated (fit) and theoretical (th) probability density.

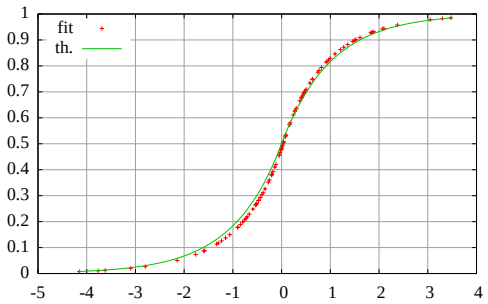

Figure 2: Estimated (fit) and theoretical (th) probability distribution.

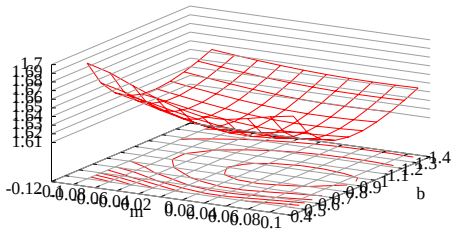

Figure 3: Negative reduced log-likelihood as a function of m and b.

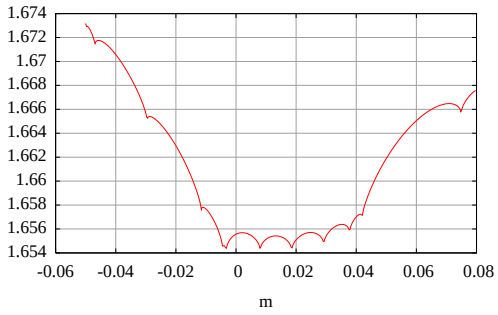

Figure 4: Negative reduced log-likelihood as a function of b.

4 Generating Subbotin Random Variates

The programs `subbogen` and `subboagen` can be used to generate pseudo random variables drawn, respectively, from a Subbotin and an asymmetric Subbotin density.

4.1 subbogen

`subbogen` generate a sequence of pseudo-random variables from a Subbotin density and print them to standard output, in ASCII format, separated by a newline character. The parameters of the density can be specified as option on the command line or can be read from standard input.

Usage: `subbogen` [options]

where [options] can be

'`-N num`' generate a number '`num`' of independent random variables.

'`-m M`' set the value of the parameter m equal to '`M`'.

'`-a A`' set the value of the parameter a equal to '`A`'.

'`-b B`' set the value of the parameter b equal to '`B`'.

'`-R seed`' set the seed of the random number generator to '`seed`'.

'`-i`' read the density parameter from the standard input. They should be provided in the order b,a,m.

'`-h`' print a short help

4.2 subboagen

`subboagen` generate a sequence of pseudo-random variables from an asymmetric Subbotin density and print them to standard output, in ASCII format, separated by a newline character. The parameters of the density can be specified as option on the command line or can be read from standard input.

Usage: `subbogen` [options]

where [options] can be

'`-N num`' generate a number '`num`' of independent random variables.

'`-m M`' set the value of the parameter m equal to '`M`'.

'`-a A`' if '`A`' is a single number, set the values of both al and ar equal to '`A`'. If '`A`' is a couple of comma separated numbers, the first is used to set al, the second to set ar.

'`-b B`' if '`B`' is a single number, set the values of both bl and br equal to '`B`'. If '`B`' is a couple of comma separated numbers, the first is used to set bl, the second to set br.

'`-R seed`' set the seed of the random number generator to '`seed`'.

'`-i`' read the density parameter from the standard input. They should be provided in the order b,a,m.

'`-h`' print a short help

5 Fitting the Subbotin Density

The programs `subbofit` and `subboafit` try to find the Subbotin density or the asymmetric Subbotin density that better fit a user-supplied dataset. These programs obtain their estimation via numerical minimization of the reduced (negative) log-likelihood (see Section 2.2 [Parameters Estimation], page 3).

The program `subbolafit` fit the same asymmetric Subbotin density fitted by `subboafit`, but in the particular case in which the two scale parameters are set equal, i.e. $a_l=a_r$.

Due to the non-analytic nature of the Subbotin log-likelihood function, both in the symmetric and asymmetric case, a straightforward minimization of that function can result dangerous. Instead, the programs use a multi-steps approach.

5.1 Fitting the Subbotin density in the symmetric case

In the symmetric case, `subbofit` obtains a first guess of the parameters values using the method of moments. Then it refines this first guess performing an unconstrained minimization of the log-likelihood function. Finally it splits the minimization procedure inside different subdomains whose interior constitute an analyticity region for the log-likelihood function and that together constitute a neighborhood covering of the previously found values. By comparing the various local minima in these domains, the program finds the global minimum. Let's analyze the procedure in more details.

5.1.1 Method of Moments

The first step of the fitting procedure consists in the estimation of the values of the parameters based on the method of moments, i.e. using the sample mean M_0 , the sample variance M_2 and the sample absolute deviation $M_{|1|}$ and following the procedure outlined in see Section 2.2 [Parameters Estimation], page 3. Let (b_0, a_0, m_0) be the parameters values obtained with the method of moments.

5.1.2 Unconstrained Minimization

In the second step we run an unconstrained minimization procedure on the reduced negative log likelihood function L (see Section 2.2 [Parameters Estimation], page 3), using as starting point the couple (b_0, m_0) previously determined with the method of moments.

In the case in which the value of m is provided on the command line (using the option '`-m`', see Section 5.4 [subbofit], page 11) the search for the minimum is reduced to a one dimensional problem. Moreover, in this case the function L is always analytical and the unconstrained minimization solve the problem finding the global minimum.

In general, when the value for m is not provided, due to the non-analyticity of the object function, the unconstrained minimization procedure does not generate reliable results and it is simply used as a first rough estimate of the solution. Denote with (b_1, a_1, m_1) the set of values obtained with the present procedure.

5.1.3 Minimization on Intervals

If $b<1$ the reduced (negative) log-likelihood function L becomes not differentiable when $m = x_j$ for $j \in \{1, \ldots, N\}$ i.e. when the parameter m takes the value of one of the observations.

The method of *Interval Constrained Minimization* try to overcome this problem by evaluating the function L only in domains where it is analytical. More specifically, one searches for the minimum inside any compact interval $[x_j, x_{j+1}]$. In this way a list of *local* minima, one for each interval, is produced. The minimization inside each interval is performed on a "smoothed"

version of L obtained with a change of variables, in such a way that the first derivative results well defined but the number and location of minima remains unaffected. Once the local minima are computed inside all the intervals, the local minimum associated with the smallest value of the function L constitutes the global minima, i.e. the point one was looking for.

The algorithm actually implemented does not apply this straightforward procedure, because the execution of a constrained minimization problem for each interval can become too expensive when the size of the sample increases. Instead, a search algorithm is implemented on the set of these intervals. Initially the minimization problem is solved on a small group of intervals surrounding the point m_1 . This initial set is enlarged progressively if new global minima are found. When for a given number of steps no new global minima appear, the search is stopped.

5.2 Fitting the Subbotin density in the asymmetric case

In the asymmetric case it is impossible to implement a simple and direct approach for the density estimation based on the method of moments. Even if the method is still valid in principle, its application requires the solution of a 5-dimensional system of non-linear equations. This strongly reduces its attractiveness as a first guess generator.

Another difference with respect to the symmetric case is the practical inconvenience of reducing the number of variables in the log-likelihood function. Indeed, in the asymmetric case it is again possible to use suitable first order conditions to remove the explicit dependence of the log-likelihood function on some variables, namely al and ar. However, this method would lead to an expression for the reduced log-likelihood that, despite the reduction of the number of independent variables form 5 to 3, appears extremely complicated and computationally more demanding.

For the above reasons, in order to obtain the asymmetric Subbotin fit, the program `subboafit` starts with an unconstrained minimization of the log-likelihood function over all its 5 parameters (for the expression of the log-likelihood function see see Section 2.1 [The Subbotin Families of Densities], page 2). Then the programs proceeds with an interval-constrained minimization, similar to the one performed by `subbofit`, whit the parameter m limited inside a compact interval formed by two consecutive observations of the dataset See Section 5.1.3 [Minimization on Intervals], page 10. A search algorithm is implemented on the local minima found inside these intervals to find the global minimum.

5.3 Fitting the asymmetric Laplace density

For the asymmetric Laplace a global maximization of the reduced, one paramter, log-likelihood to find the value of the parameter m. The parameters a_l and a_r are then computed using the formula provided in see Chapter 2 [Overview], page 2.

5.4 subbofit

`subbofit` fits a Subbotin density on a set of data provided as standard input. The method used to estimate the density parameters and the ouput format can be set with command-line options.

Usage: `subbofit` [options list] [files list]

where [files list] is a list of files containing the observations on which the density is estimated. If [files list] is empty, the program reads the data from standard input. The possible options in [options list] are

‘`-M method`’

choose a fitting method. If ‘`method`’=0 uses maximum likelihood estimation, if ‘`method`’=1 uses method of moments See Section 2.2 [Parameters Estimation], page 3.

'`-O output`'
: choose an output format. If '`output`'=0 print to the standard ouput the estimated b, a, m and reduced negative log-likelihood. If '`output`'=1 prints each datapoint with the associated distribution function. If '`output`'=2 prints each datapoint with the associated density.

'`-V verbosity`'
: choose a verbosity level. If '`verbosity`'=0 print only the final results. More information can be obtained increasing the verbosity level.

'`-m M`'
: set the parameter m equal to '`M`'. The optimization is performed only on the two remaining parameters.

'`-G step,tol,iter, eps, algo`'
: set the parameter for the global numerical optimization

'`-I step,tol,iter, eps, algo`'
: set the parameter for the interval-constrained numerical optimization.

'`-s steps`'
: set to '`steps`' the step of the discrete optimization algorithm on the set of local minima obtained with interval-constrained optimization.

'`-h`'
: print a short help with the default values for the various parameters

The parameters of the numerical optimization are set using a comma separated list of 5 parameters. Empty fields leave the default unchanged. The meaning of the various parameter is as follows:

'`step`'
: initial step size of the searching algorithm

'`tol`'
: tolerance of the line search algorithm

'`iter`'
: maximum number of iterations

'`eps`'
: gradient tolerance: the search is stopped when `||`gradient`||` < eps

'`algo`'
: choose the optimization method: 0 Fletcher-Reeves, 1 Polak-Ribiere,2 Broyden-Fletcher-Goldf, 3 Steepest descent

5.5 subboafit

`subboafit` fits an asymmetric Subbotin density on a user supplied set of data. The ouput format can be set with command-line options.

Usage: `subboafit` [options list] [files list]

where [files list] is a list of files containing the observations on which the density is estimated. If [files list] is empty, the program reads the data from standard input. The possible options in [options list] are

'`-O output`'
: choose an output format. If '`output`'=0 print to the standard ouput the estimated b, a, m and reduced negative log-likelihood. If '`output`'=1 prints each datapoint with the associated distribution function. If '`output`'=2 prints each datapoint with the associated density.

'`-V verbosity`'
: choose a verbosity level. If '`verbosity`'=0 print only the final results. More information can be obtained increasing the verbosity level.

'`-m M`'
: set the parameter m equal to '`M`'. The optimization is performed only on the two remaining parameters.

‘`-G step,tol,iter, eps, algo`’
set the parameter for the global numerical optimization

‘`-I step,tol,iter, eps, algo`’
set the parameter for the interval-constrained numerical optimization.

‘`-s steps`’
set to ‘`steps`’ the step of the discrete optimization algorithm on the set of local minima obtained with interval-constrained optimization.

‘`-h`’ print a short help with the default values for the various parameters

The parameters of the numerical optimization are set using a comma separated list of 5 parameters. The meaning of the parameters and the syntax is the same as for `subbofit` (see Section 5.4 [subbofit], page 11)

5.6 subbolafit

`subbolafit` fits a (less) asymmetric density on a user supplied set of data. The less asymmetric density is the asymmetric Subbotin density (see Section 2.1 [The Subbotin Families of Densities], page 2) with the two scale parameters equal, $al = ar$ The ouput format can be set with command-line options.

Usage: `subbolafit` [options list] [files list]

where [files list] is a list of files containing the observations on which the density is estimated. If [files list] is empty, the program reads the data from standard input. The possible options in [options list] are

‘`-O output`’
choose an output format. If ‘`output`’=0 print to the standard ouput the estimated b, a, m and reduced negative log-likelihood. If ‘`output`’=1 prints each datapoint with the associated distribution function. If ‘`output`’=2 prints each datapoint with the associated density.

‘`-V verbosity`’
choose a verbosity level. If ‘`verbosity`’=0 print only the final results. More information can be obtained increasing the verbosity level.

‘`-m M`’ set the parameter m equal to ‘`M`’. The optimization is performed only on the two remaining parameters.

‘`-G step,tol,iter, eps, algo`’
set the parameter for the global numerical optimization

‘`-I step,tol,iter, eps, algo`’
set the parameter for the interval-constrained numerical optimization.

‘`-s steps`’
set to ‘`steps`’ the step of the discrete optimization algorithm on the set of local minima obtained with interval-constrained optimization.

‘`-h`’ print a short help with the default values for the various parameters

The parameters of the numerical optimization are set using a comma separated list of 5 parameters. The meaning of the parameters and the syntax is the same as for `subbofit` (see Section 5.4 [subbofit], page 11)

5.7 laplaafit

`laplaafit` fits an asymmetric Laplace density on a user supplied set of data. The asymmetric Laplace density is a special asymmetric Subbotin density (see Section 2.1 [The Subbotin Families of Densities], page 2) with the two tail parameters equal to one. The ouput format can be set with command-line options.

Usage: `laplaafit` [options list] [files list]

where [files list] is a list of files containing the observations on which the density is estimated. If [files list] is empty, the program reads the data from standard input. The possible options in [options list] are

'`-O output`'
: choose an output format. If '`output`'=0 print to the standard ouput the estimated b, a, m and reduced negative log-likelihood. If '`output`'=1 prints each datapoint with the associated distribution function. If '`output`'=2 prints each datapoint with the associated density.

'`-V verbosity`'
: choose a verbosity level. If '`verbosity`'=0 print only the final results. More information can be obtained increasing the verbosity level.

'`-m M`'
: set the parameter m equal to '`M`'. The optimization is performed only on the two remaining parameters.

'`-G step,tol,iter, eps, algo`'
: set the parameter for the global numerical optimization

'`-h`'
: print a short help with the default values for the various parameters

The parameters of the numerical optimization are set using a comma separated list of 5 parameters. The meaning of the parameters and the syntax is the same as for `subbofit` (see Section 5.4 [subbofit], page 11)

6 Exploring the symmetric Subbotin Log-Likelihood

The possibility of visually explore the reduced negative log-likelihood of a given dataset becomes very useful when the fitting program produces some error and you want to directly verify if the found minima is actually the global minima you were looking for. The program `subboshow` provides an easy way of performing such a graphical exploration. It is essentially intended to be used inside the `gnuplot` graphic environment. The format of the output has been conceived to be sent to `gnuplot`'s `plot` or `splot` commands. For an example of use See Section 3.2 [Graphic Tutorial], page 7.

6.1 subboshow

The program `subboshow` allows the visual exploration of the Subbotin reduced (negative) log-likelihood of an user-supplied dataset.

Usage: `subboshow` [options list] [files list]

where [files list] is a list of files containing the observations with which the log-likelihood is computed. If [files list] is empty, the program reads the data from standard input. The possible options in [options list] are

'`-M num`' print the reduced negative log-likelihood for '`num`' equally spaced values of the parameter m.

'`-B num`' print the reduced negative log-likelihood for '`num`' equally spaced values of the parameter b.

'`-m min,max`'
print the reduced negative log-likelihood for values of m in the range min, max.

'`-b min,max`'
print the reduced negative log-likelihood for values of b in the range min, max.

'`-h`' print a short help and default parameters values

7 Installation

You require the GSL (Gnu Scientifi Library) to be present on your system in order to install `SUBBOTOLS`. Check that these libraries are properly installed on your system before proceeding to the installation of the present package. For informations about GSL, including installation instructions, check `http://www.gnu.org/software/gsl/`.

SUBBOTOLS comes with a `configure` program in the GNU style. Installation can be as simple

```
tar xvzf subbotools-[version].tar.gz
cd subbotools-[version]
./configure
make
make install
```

where `[version]` stands for a number indicating the package version.

The commands above install the `SUBBOTOLS` executable in '`/usr/local/bin`' and the documentation, in the form of info files, in '`/usr/local/info`'.

To obtain pdf or html version of the manual, after the configuration, do

```
cd doc
make pdf
make html
```

For more detailed installation instruction see the file '`INSTALL`'.

8 License and Copyright

SUBBOTOOLS is copyright © 2002-2003 Giulio Bottazzi.

SUBBOTOOLS is a collection of free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This package is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this package; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA. You can also find the GPL on the GNU web site.

In addition, I kindly ask you to acknowledge SUBBOTOOLS and its author in any program or publication in which you use SUBBOTOOLS. (You are not *required* to do so; it is up to your common sense to decide whether you want to comply with this request or not.)

Concept Index

A

Asymmetric Subbotin probability density 2

B

Bugs Report 1

C

Central moments 2
Command Line 6

E

Examples 6

F

Fitting 10
Fitting, Laplace asymmetric 11
Fitting, Subbotin asymmetric 11
Fitting, Subbotin symmetric 10

G

`gnuplot` 5
Graphics 5
GSL 1

H

Howto 6

I

Input 5
Introduction 1

L

Likelihood 3

M

Method of Moments 10
Minimization on intervals 10
Minimization, unconstrained 10

O

Output 5
Overview 2

P

Parameters 2
`plotutils` 5
Probability density 2

R

Reduced log-likelihood 3

S

Subbotin central moments 2
Subbotin likelihood 3
Subbotin parameters 2
Subbotin probability density 2
Subbotin reduced log-likelihood 3

T

Tutorial 6

U

Unconstrained Minimization 10